

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ  
ФЕДЕРАЦИИ**  
Федеральное государственное автономное образовательное учреждение  
высшего образования  
**«Национальный исследовательский  
Нижегородский государственный университет им. Н.И. Лобачевского»  
(ННГУ)**

**Институт информационных технологий, математики и механики**

Направление подготовки: «Прикладная математика и информатика»  
Магистерская программа: «Вычислительные методы и суперкомпьютерные  
технологии»

Образовательный курс «Методы глубокого обучения для решения задач  
компьютерного зрения»

**ОТЧЕТ**  
по лабораторной работе №3

**«Разработка сверточной нейронной сети с помощью  
библиотеки MXNet»**

**Выполнили:**  
студенты группы 381603м4  
Кривоносов Михаил  
Герасимов Дмитрий  
Уваров Денис

Нижний Новгород  
2017

## Оглавление

Цель работы .....	3
Задачи .....	3
Тренировочные и тестовые данные .....	4
Схемы конфигураций сетей .....	5
Выводы .....	8

## **Цель работы**

Построить архитектуру сверточной нейронной которая позволила бы решать практическую задачу с высоким показателем качества.

## **Задачи**

- 1) Разработать нескольких архитектур сверточных нейронных сетей, который обрабатывается библиотекой MXNet.
- 2) Обучить и провести тестирование нейронных сетей

## Тренировочные и тестовые данные

Для предварительной обработки данных использован скриптовый язык Python. Реализован скрипт, в котором начальный трехмерный массив данных преобразуется в хронологическую развертку для каждого пикселя с учетом его окрестности (поле  $5 \times 5$ , элементы, выходящие за пределы исходного поля, отождествляем с нулем), тем самым для каждого пикселя получаем трехмерную подматрицу размера  $5 \times 5 \times 300$ . Тем самым получается выборка из 262144 элементов, где в качестве каждого элемента указанная подматрица.

Затем данные случайным образом делятся на обучающую и тестовую выборки в отношении 67% к 33%. Так как полученная выборка достаточно большого объёма, то необходимо уменьшить размер выборки вдвое. Полученные массивы  $X_{train}$ ,  $X_{test}$ ,  $Y_{train}$ ,  $Y_{test}$  передаются в нейронной сети для дальнейшей работы с ними.

**Размер исходных данных:** трёхмерная матрица размером  $512 \times 512 \times 300$ .

## Схемы конфигураций сетей

Рассмотрим три различных конфигурации сверточной нейронной сети.  
Коэффициент скорости обучения  $\eta = 0.007$

### 1. Первая конфигурация

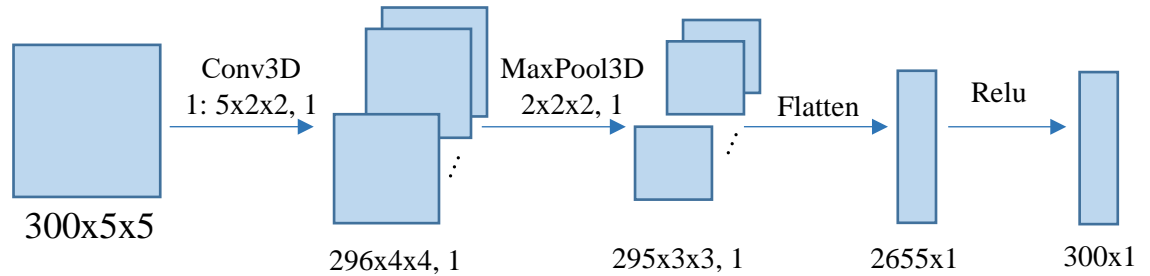


Рисунок 1. Схема нейронной сети

В качестве функции потерь используется **LogisticLoss**

**Результат обучения:**

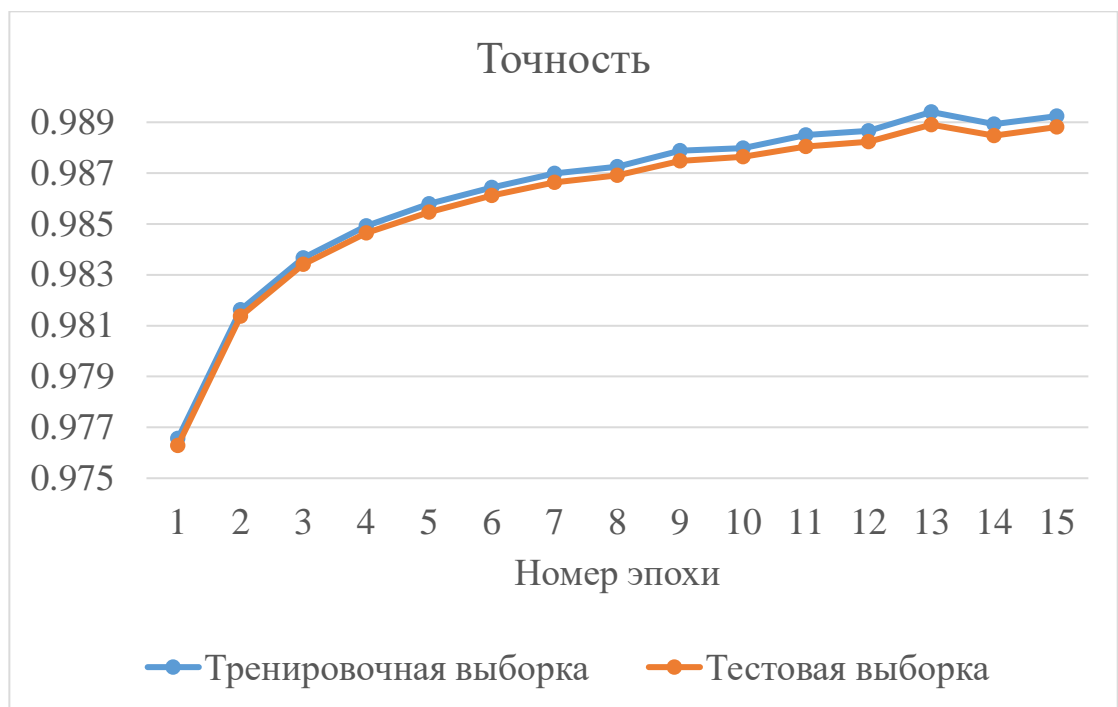


Рисунок 2. График значения точности в зависимости от номера эпохи

## 2. Вторая конфигурация.

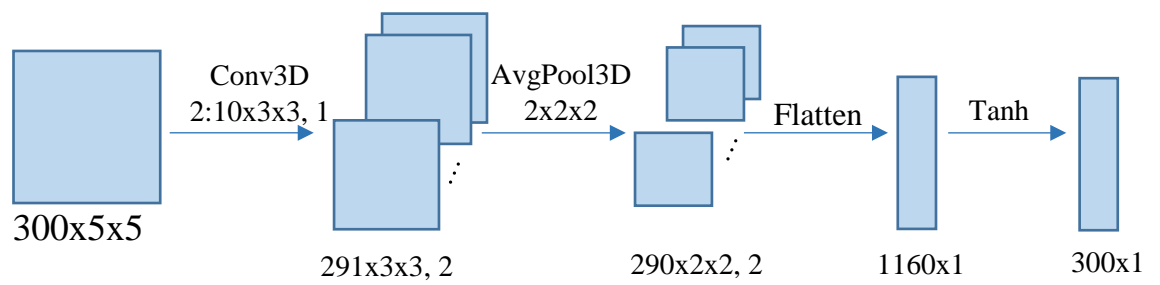


Рисунок 3. Схема нейронной сети

В качестве функции потерь используется **LogisticLoss**

**Результат обучения:**

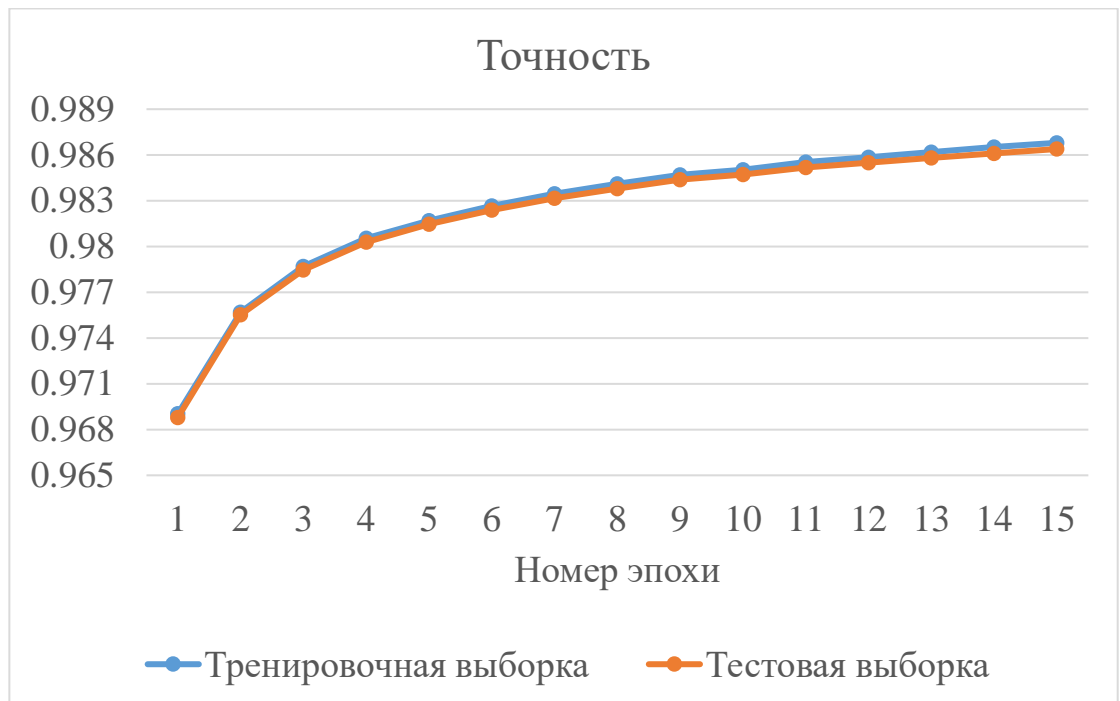


Рисунок 4. Схема нейронной сети

### 3. Третья конфигурация.

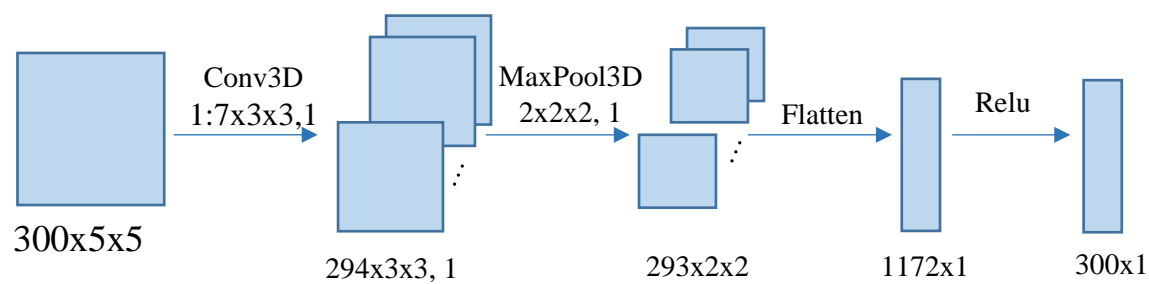


Рисунок 5. Схема нейронной сети

В качестве функции потерь используется **HingeLoss**

**Результат обучения:**

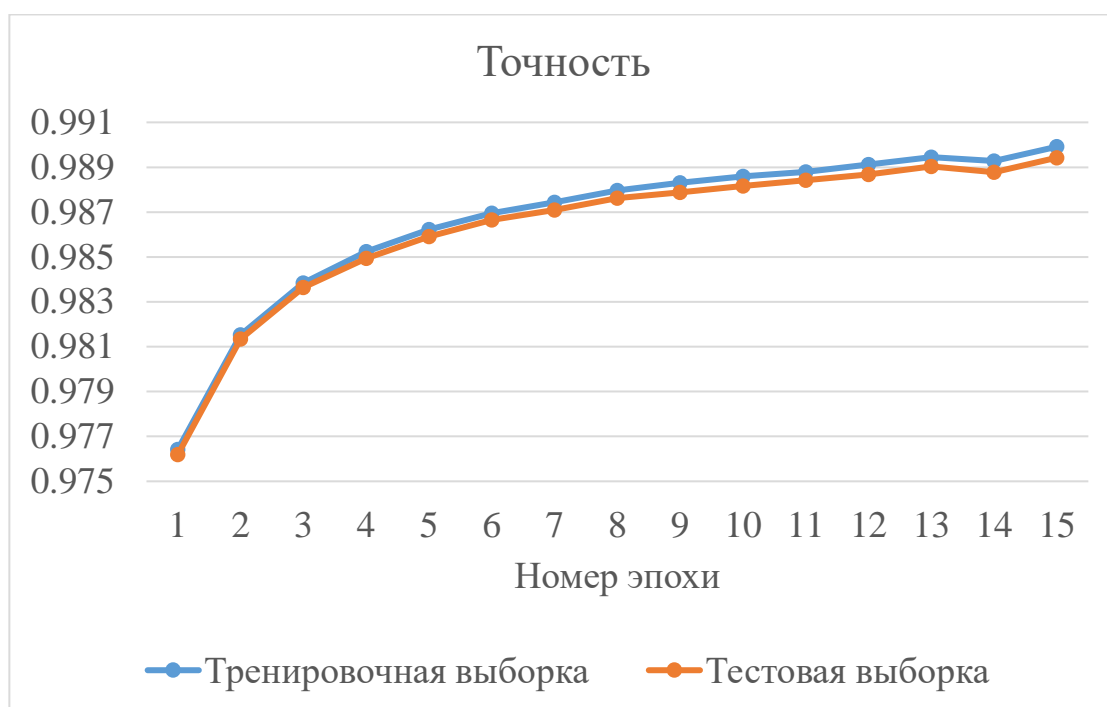


Рисунок 6. График значения точности в зависимости от номера эпохи

## **Выводы**

Применение сверточных нейронных сетей на выбранных входных данных не дало прироста точности, по сравнению с полностью связанной нейронной сетью. С ростом числа эпох точность увеличивается.